

### **IN THE CLAIMS:**

Applicants elect the Group II claims in response to the Restriction Requirement dated April, 9, 2004. The Group II claims are 6 through 10, 17 and 20, all of which are reproduced below.

6. A method for performing variable time processes in parallel on a plurality of ordered incoming entities, on a plurality of processors, the method comprising:  
  
assigning each of the plurality of incoming entities to one of the plurality of processors;  
  
pre-processing each of the plurality of ordered incoming entities to establish an order of the plurality of ordered incoming entities; and  
  
processing each of the plurality of ordered incoming entities on the corresponding one of the plurality of processors to which it is assigned.
7. The method of claim 6, wherein the plurality of ordered incoming entities comprise a plurality of packets in a network.
8. The method of claim 7 wherein the processing comprises:  
  
extracting information from a header of each of the plurality of incoming packets;  
  
hashing the extracted information for each of the incoming packets;  
  
storing the hashed information; and  
  
responsive to a new packet being received, comparing the hash of the extracted information for the new packet with the stored information.

9. The method of claim 6, wherein the assigning comprises:
- selecting one of the plurality of processors which is free to process one of the plurality of ordered incoming entities at the time that the one of the plurality of ordered incoming entities is received.
10. The method of claim 6, wherein the pre-processing comprises:
- determining whether a first in the plurality of ordered incoming entities is currently being processed at the time when a subsequent one of the plurality of ordered incoming entities is received;
- responsive to determining that the first in the plurality of ordered incoming entities is currently being processed:
- not starting processing of the subsequent one of the plurality of ordered incoming entities;
- redetermining at a later time whether the first in the plurality of ordered incoming entities is currently being processed; and responsive to determining that the first in the plurality of ordered incoming entities is not currently being processed, starting processing of the subsequent one of the plurality of ordered incoming entities.
17. A method for performing variable time processes in parallel on a plurality of processors on a plurality of packets in a network which comprise a plurality of flows, the method comprising:
- assigning each of the plurality of packets to one of the plurality of processors;

determining whether a first in the plurality of packets comprising one of the plurality of flows is currently being processed at the time when a subsequent one of in the plurality of packets comprising the one of the plurality of flows is received;

responsive to determining that the first in the plurality of packets is currently being processed:

not starting processing of the subsequent one of the plurality of packets;

redetermining at a later time whether the first in the plurality of packets is currently being processed;

responsive to determining that the first in the plurality of packets is not currently being processed, starting processing of the subsequent one of the plurality of packets;  
and

processing each of the plurality of packets on the corresponding one of the plurality of processors to which it is assigned.

20. An electronically readable medium storing a program for permitting a computer to perform a method for performing variable time processes in parallel on a plurality of processors on a plurality of packets in a network which comprise a plurality of micro-flows, the method comprising:

assigning each of the plurality of packets to one of the plurality of processors;

determining whether a first in the plurality of packets comprising one of the plurality of micro-flows is currently being processed at the time when a subsequent one of in the plurality of packets comprising the one of the plurality of micro-flows is received;

responsive to determining that the first in the plurality of packets is currently being processed:

not starting processing of the subsequent one of the plurality of packets;

redetermining at a later time whether the first in the plurality of packets is currently being processed;

responsive to determining that the first in the plurality of packets is not currently being processed, starting processing of the subsequent one of the plurality of packets;  
and

processing each of the plurality of packets on the corresponding one of the plurality of processors to which it is assigned.